

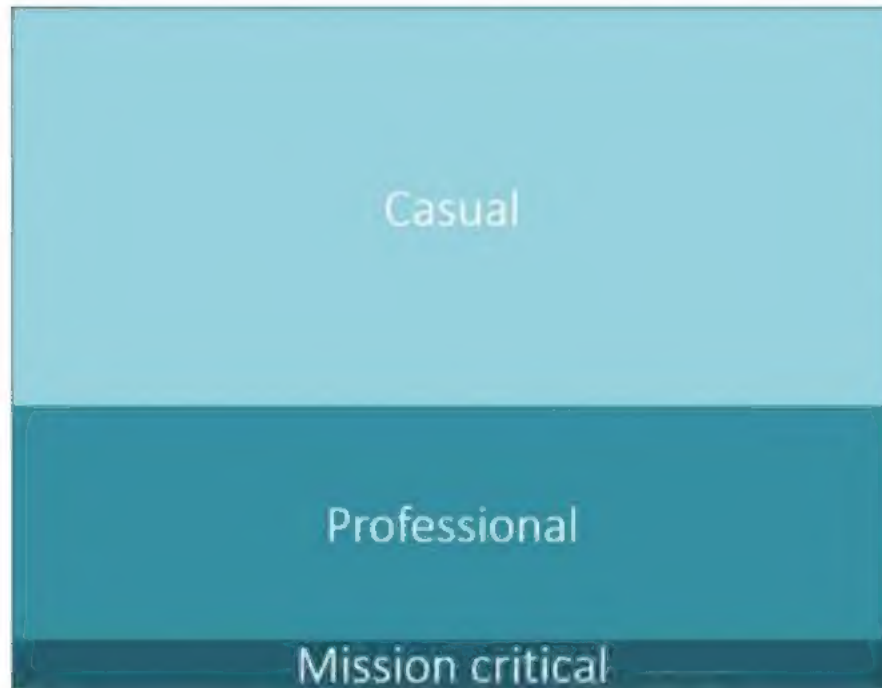
# Verified correctness of critical programs

Chris Hawblitzel (Operating Systems Group)

Rustan Leino (Research in Software Engineering)

Bryan Parno (Security and Privacy Research Group)

# Who would verify their programs?



Target verification applications:



Research direction

# Verification tools

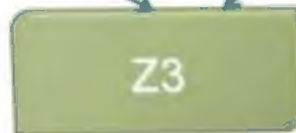
Programming language



Intermediate  
verification language



Decision procedures



# Outline

Small demo

CloudMake verification

IronClad verification

# Demo

Log

# CloudMake

Modern *make* utility

Functional subset of TypeScript

*exec* construct – calls out to compilers, linkers, etc.

Algorithm correctness

Formalized as a program in Dafny

*Interpreter for language*

*Axioms for *exec**

Correctness properties proved

*Parallel builds are correct*

*Cache is consistent*

*Cached behavior = clean-build behavior*



# Demo

CloudMake [joint work with Maria Christakis and Wolfram Schulte]

# ***The Ironclad Project:*** Full Verification of Security-Sensitive Services in Dafny



Chris Hawblitzel



Jon Howell



Jay Lorch



Bryan Parno



Brian Zill



Arjun Narayan



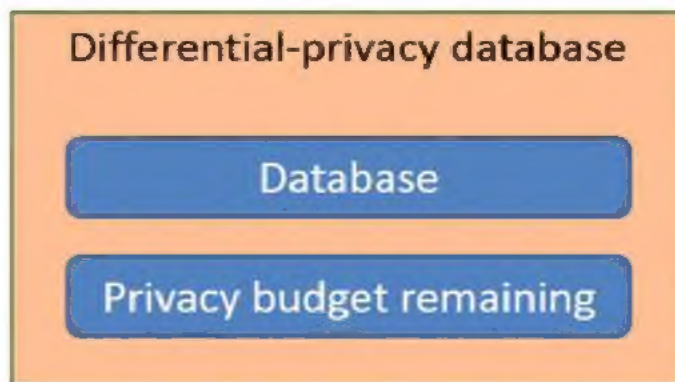
Danfeng Zhang



## Two **ironclad examples**: append-only audit log and differential privacy



Entries cannot be removed.

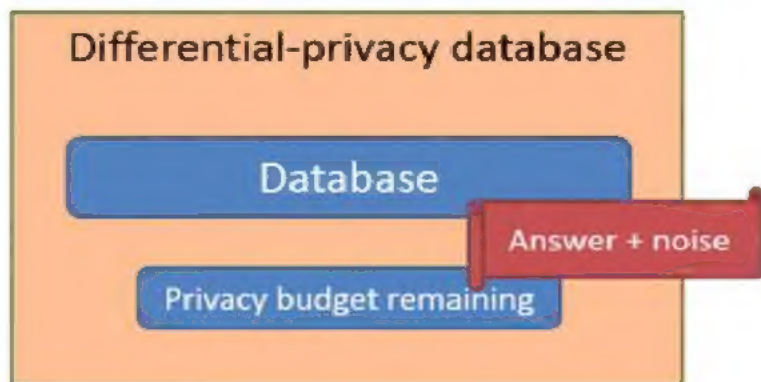


Private data revealed only according to policy.

## Two **ironclad examples**: append-only audit log and differential privacy

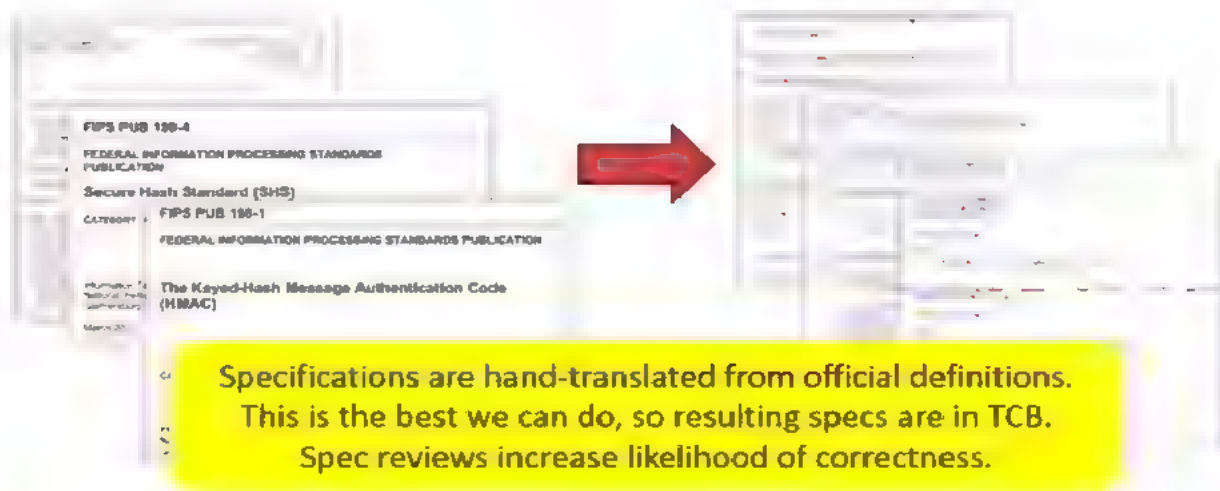


Entries cannot be removed.



Private data revealed only according to policy.

Security-sensitive services need cryptography, so we built a **crypto library**.

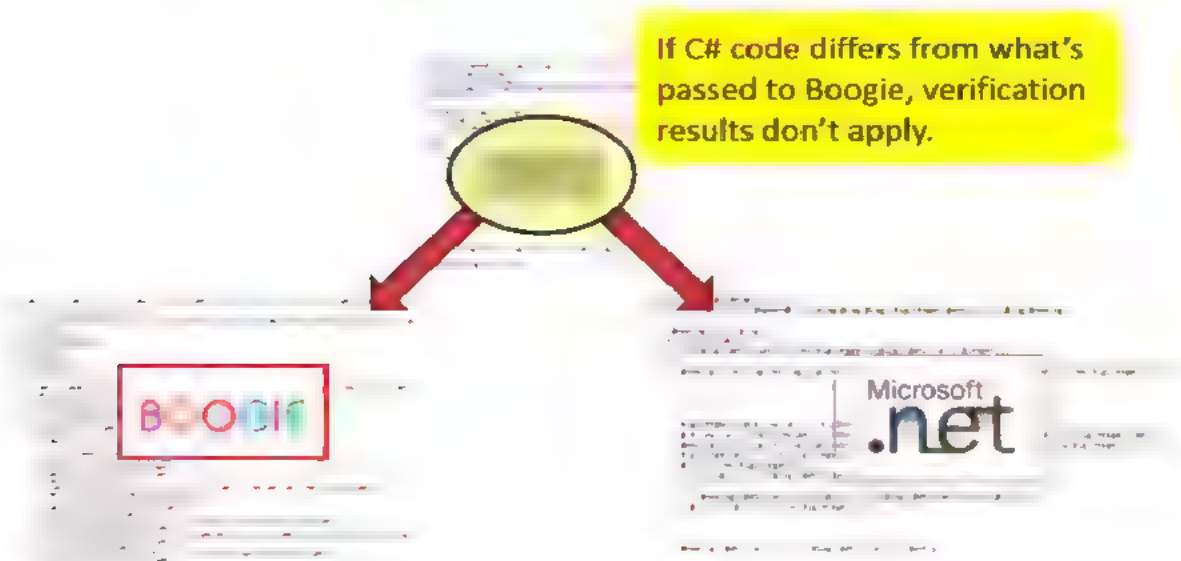


Ironclad apps written in Dafny  
...but standard **Dafny toolchain** includes large TCB.



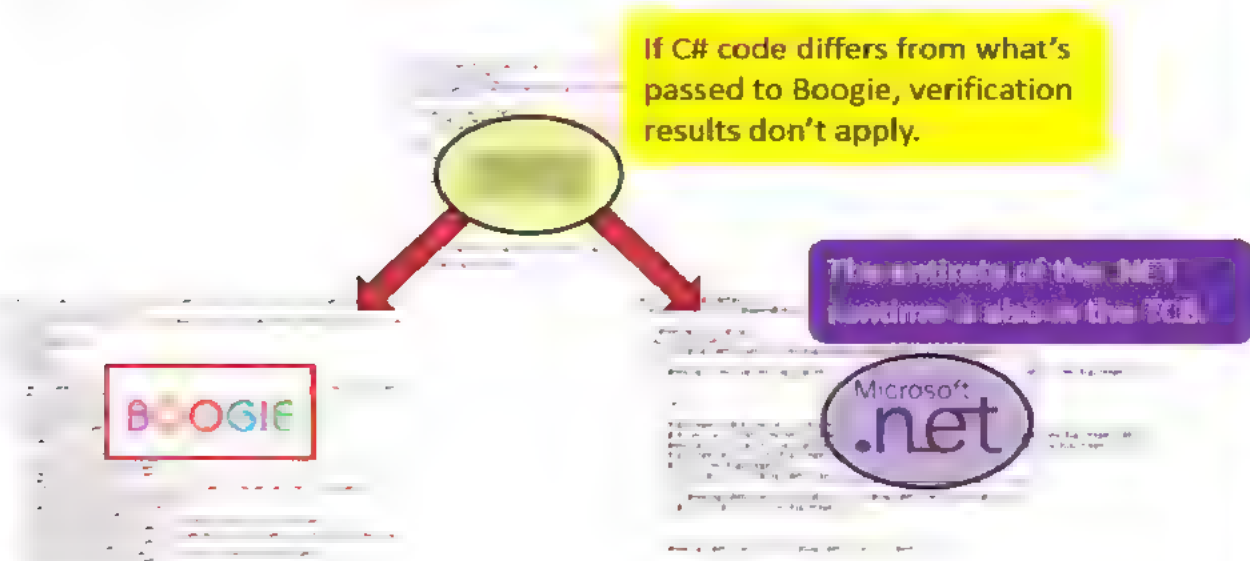
# Ironclad apps written in Dafny

...but standard **Dafny toolchain** includes large TCB.

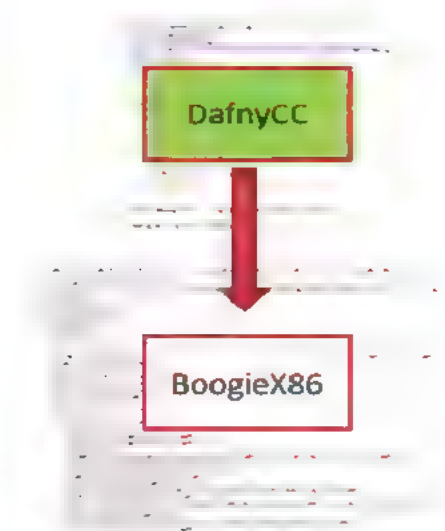


# Ironclad apps written in Dafny

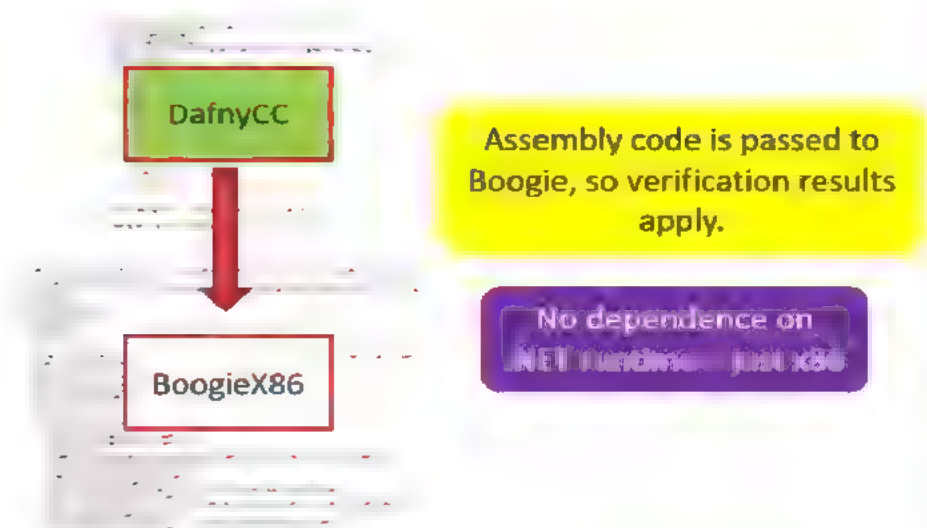
...but standard **Dafny toolchain** includes large TCB.



Our **new Dafny compiler** substantially reduces the TCB.



Our **new Dafny compiler** substantially reduces the TCB.





# DafnyCC demo

2

# DafnyCC features

- Supports large subset of Dafny language:
  - Expressions: add, sub, bitwise, booleans, function calls, ...
  - Statements: assignment, if/else, while, ...
  - Types: int, bool, array-of-int, datatypes
- No object-oriented features
- Linear scan register allocator
- Not an optimizing compiler!

# Verve: a verifiably safe OS

"every assembly  
language instruction  
checked for safety"

Kernel/Apps

Small Operating System + Applications (Dafny)

DafnyCC

Boogie

Nucleus

Verified  
Garbage  
Collector

Verified  
Threads

Verified  
Interrupt  
Handlers

Verified  
Device  
Interface

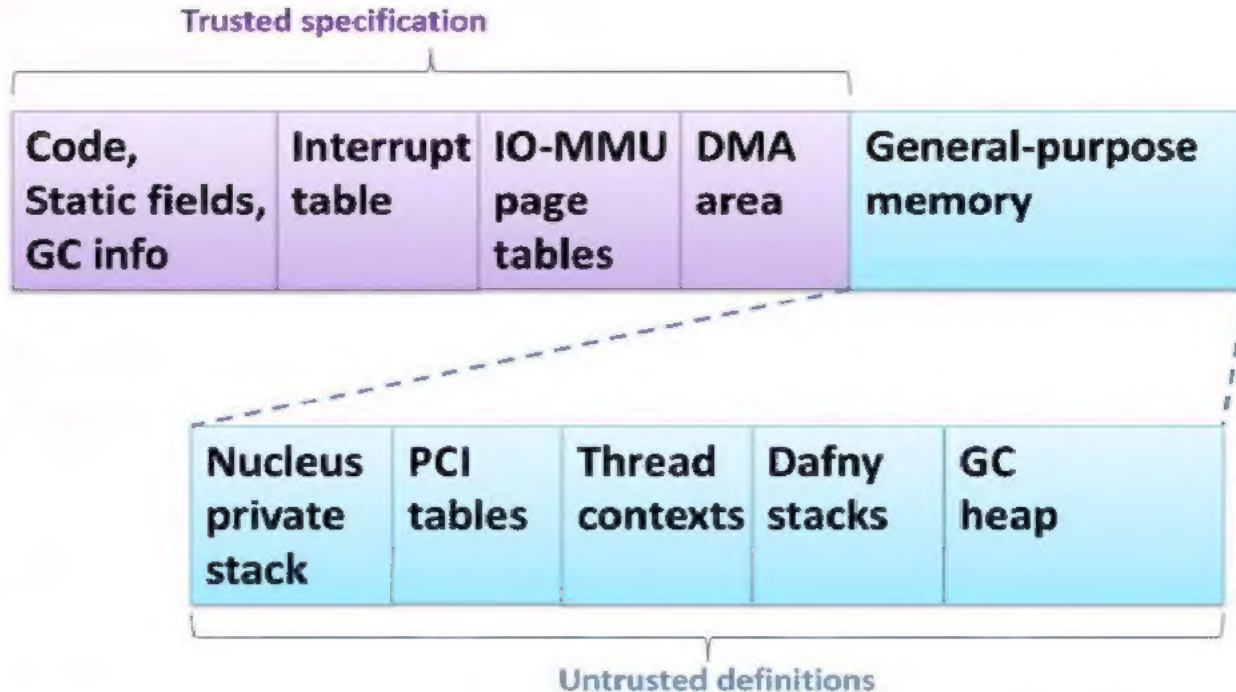
Verified  
Startup

Verify  
safety &  
correctness  
with  
Boogie/Z3

Boot Loader

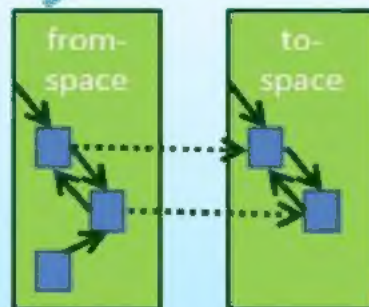
x86 Hardware

# Memory management



# Memory management

```
...  
&& (forall i:int::{T(i)}  
  T(i) && Fi <= i && i < Fk && r1[i] != NO_ABS &&  
  (IsFwdPtr(gcMem[i + 4])) ==>  
    Pointer(r2, gcMem[i + 4] - 4, r1[i])  
  && AlignedHeapAddr(i + 4)  
  && word(gcMem[i + 4]))  
...
```



Untrusted definitions

# Ironclad progress so far

- Verve
- Libraries
  - Big integers and rationals
  - Crypto: SHA-1, SHA-256, HMAC, RSA
  - Utilities for manipulating bytes, words, sequences, arrays
  - Math
- Drivers
  - TPM
  - Network
- Services
  - Password vault
  - Notary
  - TrInc
  - Differentially-private database